

千葉大学工学部都市環境システム学科
卒業論文

P2P システムにおける出線数情報を用いた検索メッセージ削減法

平成 17 年 2 月提出

指導教員 塩田 茂雄 助教授

環境システム講座
情報システム評価教育研究分野

石原 田久

目次

第1章	序論.....	2
第2章	P2P ネットワーク.....	3
2.1	P2P とは.....	3
2.2	システム形態.....	3
2.3	HybridP2P 型.....	4
2.4	PureP2P 型.....	4
第3章	研究概要.....	6
3.1	無駄なクエリーの発生要因と提案.....	6
3.2	無駄なクエリーの発生の検証.....	7
第4章	リンクカットの手法.....	8
4.1	ランダムリンクカット.....	8
4.2	出線数ベースリンクカット A.....	9
4.3	出線数ベースリンクカット B.....	10
第5章	タイムアウト処理.....	11
5.1	タイムアウト処理の原理.....	11
5.2	タイムアウト時間の設定.....	12
第6章	シミュレーション実験.....	13
6.1	BRITE.....	13
6.1.1	Waxman モデル.....	14
6.1.2	BA モデル.....	14
6.2	ネットワークの構築.....	15
6.2.1	IP ネットワークの構築.....	15
6.2.2	距離の書き換え.....	16
6.2.3	P2P ネットワークの構築.....	18
6.2.4	P2P ネットワークのピア間の距離の設定.....	18
6.2.5	シミュレーション結果データ.....	19
6.3	シミュレーション結果.....	20
6.3.1	シミュレーションに用いたデータ.....	20
6.3.2	各 Case の結果.....	22
第7章	シミュレーション結果による考察.....	27
第8章	まとめ・結論.....	31
	謝辞.....	34
	参考文献.....	34

第1章 序論

今日、ある情報を得るためにインターネットで検索をするという方法があるが、その検索方法は主に2種類ある。1つは google のような Web に代表される集中型の検索方法である。集中型の検索では、特定のホストにインデックスが集中し、そこでのみ検索を行う。もう1つは、分散型の検索方法である。分散型では、インデックスがネットワーク上に分散し、検索はネットワーク全体で行われる。

分散型には、さらに、人間によって管理されたネットワークと管理されないネットワークとがある。管理されたネットワークとは、Domain Name System のようにノード情報、階層構造を人間の手によって管理するネットワーク形態である。管理されたネットワークでは、検索経路が管理されているので、動的に経路を決定する際の無駄なクエリーを減らせる長所がある。

一方、管理されないネットワークとは Peer-to-Peer (P2P) 型のネットワークを指す。Gnutella に代表される P2P 型のネットワークでは完全分散型のネットワークを形成している。完全分散型のネットワークでは各ノードが互いに対等であり、ノード情報は人間に管理されない。一つのノードに情報が集中しないため、一つのノードが停止してもネットワーク全体に大きな影響を与えにくく、全体的に安定している。

本稿では分散型の検索方法の中でも、P2P ネットワークについて研究したものである。従来の P2P ネットワークのコンテンツ検索方法はフラッディングと呼ばれる手法を用いている。フラッディングとはクエリーを受け取った各ピアがクエリーを送ってきたピア以外の隣接ピアすべてにクエリーをローカルブロードキャストする伝達方法である。この方法だとコンテンツ検索に要する時間はもっとも短くなる反面、同一ピアが複数の経路からクエリーを重複して受信してしまうため、検사용トラフィックに無駄が多く、ネットワークに大きな負荷を与えてしまう。この無駄なクエリーの発生を最小限にとどめようとするのが本研究の狙いである。

第2章 P2P ネットワーク

本章では、本研究の根幹となる Peer-to-Peer (P2P) ネットワークの特徴や種類について説明する。

2.1.P2P とは

Peer-to-Peer ネットワークとは、コンピューター同士を対等な立場で接続するネットワークの接続形態である。「Peer」とは「同僚」という意味を持つ言葉である。Peer-to-Peer 型ネットワークでは、接続されたコンピューターがクライアントとサーバ（サービスを提供する側とサービスを受ける側）の両方の機能を持ち相互にハードディスクなどを公開することで、データのやり取りを実行できるのが特徴である。

Peer-to-Peer 型の構内情報通信網（LAN）は 10 代程度の小規模なネットワークに適している。LAN について特別な知識がなくてもパソコン同士を接続し、ソフトウェアやプリンタの共有ネットワークが低コストで簡単に構築できる。

P2P では数十台の接続もできるが、ネットワークを管理するコンピューターがないので、オペレーティングシステム面でのセキュリティ管理が難しい欠点がある。P2P 型のネットワーク構築は、一般に数台のコンピューターが接続された環境やデータ交換が少ない場合は適しているが、中規模以上の LAN 構築では支障が多いといえる。

2.2.システム形態

P2P は大きく分けて HybridP2P 型と PureP2P 型の 2 つに分類される。HybridP2P 型は一部にサーバの思想が残っているため Hybrid と呼ばれる。サーバは存在するが、ピア同士を接続する役割のみをする。（Napstar、WinMX）

一方、PureP2P 型はサーバを必要とせず、純粋にピアのみで構成することができるため「Pure」と呼ばれる。（Gnutella、Winny、KaZaA）

PureP2P 型は完全にサーバが存在しないため、システム投資コストの低減が可能である。また、同様の理由から各クライアントは同時にサーバの役割を果たすことにより、システム負荷の分散が可能である。

2.3 .HybridP2P 型

Hybrid 型はファイルの転送に関しては P2P で行われるが、ファイルインデックスを管理する中央のサーバが存在し、検索作業はこの中央のサーバを通して行われる。各ピアは、サーバを通じてネットワークに加わる。ピアすべての情報はサーバで管理されている。検索は中央のサーバに問い合わせることで行われる。したがって、中央のサーバが停止したときには、ネットワーク全体が機能せず、P2P の特徴であるフォールト・トレランスは弱くなっている。

Hybrid P2P 型に代表される Napstar は P2P 型のネットワークを世に知らしめたが、完全な P2P ネットワークではない。Napstar では次のように検索を行う。(図 1 参照)

1. クライアントはサーバにファイルの検索内容を送る。
2. サーバはクライアントから得たファイルの検索内容を元に自分のデータベースを検索する。検索結果を送信元のクライアントに送る。
3. クライアントはサーバから送られてきた検索結果を元に、直接ほかのクライアントとファイル交換する。

ファイル検索に中央のサーバを用いるので、検索速度は速く、ネットワークの使用帯域も少なく済む長所を持っている。一方、検索をすべて中央のサーバに依存しているため、そこがボトルネックになっている欠点を持つ。

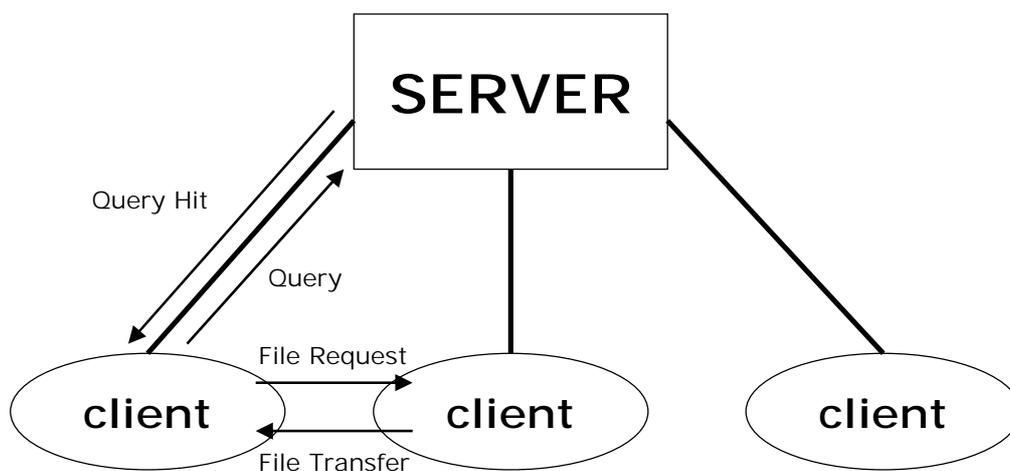


図 1.Napstar における検索

2.4.Pure P2P 型

HybridP2P とは異なり、完全分散型の P2P ネットワークをなす。このネットワークでは中心となるサーバが存在せず、各ピアが対等である。完全分散型ネットワークは単一のノードに依存しないので、あるピアがシャットダウンしてもネットワーク全体はまったく影響を受けない。

PureP2P 型に代表されるのは Gnutella である。Gnutella では次のように検索を行う。(図 2 参照)

1. あるノードは検索項目をブロードキャストする。このピアは検索時に接続している他のピアすべてにクエリーを出す。
2. クエリーを受け取ったピアは、まずそのクエリーが以前に来たものであるか調べる。同一ピアからの同一のクエリーの重複を避けるために、そのクエリーが以前きたものであれば、そのピアはそれ以上何もしない。そのクエリーが初めてくるものであれば、自分がファイルを持っているか調べ、持っていればクエリーを転送してきたピアにクエリーヒットで返す。持っていなければクエリーヒットは返さない。
3. ピアはクエリーを転送してきたピア以外の接続中のピアに Time-to-Live (TTL) を一つ減らしてから転送する。TTL とは、クエリーの転送できる回数であり、それが 0 になると、そのクエリーはそれ以上転送されない。
4. 以下、クエリーは再帰的に転送されていく。

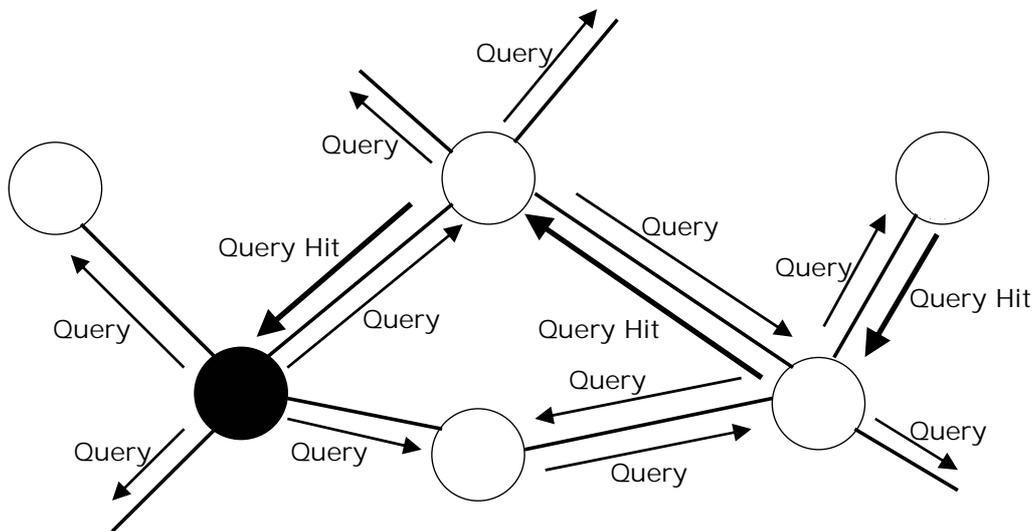


図 2.Gnutella における検索

第3章 研究概要

3.1.無駄なクエリーの発生要因と提案

無駄なクエリーが発生する場合、クエリーのリンク衝突が発生していることが多い。クエリーのリンク衝突とは、ピア間を結ぶ双方向リンク上を二つのクエリーが互いに異なる方向に行きかうケース(図3)を指す。片方向の転送ですむはずのものを双方向に転送してしまうため無駄なクエリーが増えてしまう。

本研究ではピア間を結ぶ双方向リンクを(クエリー転送に関して)いろいろな手法をもとに片方向化することにより、クエリーのリンク衝突を回避し、無駄なクエリーを削除する手法について検討した。

なお、双方向リンクを二つの片方向リンクと見た場合、「双方向リンクの片方向化」は「(いずれかの)片方向リンクの削除(カット)」と同じであるため、以下では双方向リンクの片方向化を「リンクカット」と呼ぶことにする。

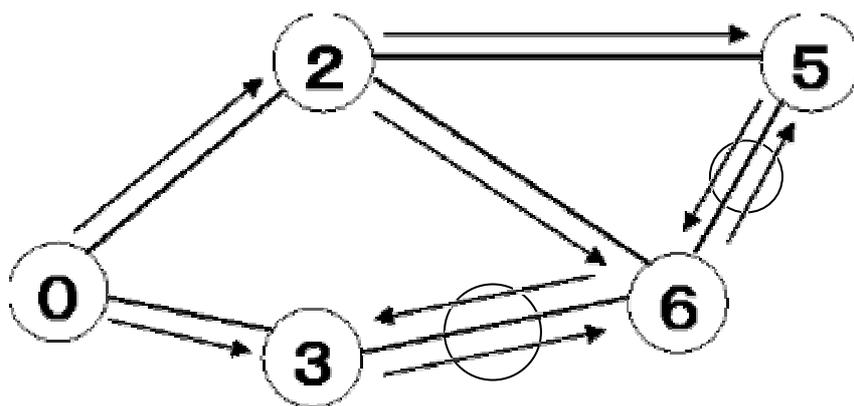


図3.クエリーのリンク衝突

3.2.無駄なクエリー発生を検証

n 個のピアと m 本のリンクから構成されるネットワークで、無駄なクエリーがどれだけ発生するのかを考えてみる。最低限全体に行き届くのに必要なクエリー数 Q_{min} は、発信元となるピアを除く全ピアに 1 つずつクエリーが受信されればよいので、

$$Q_{min} = n - 1$$

で表される。

次に各ピアのリンク数を m_i とすると、あるピア i から送信されるクエリーの数 Q_i は、クエリーが送信されてきたリンク以外のリンクすべてに送信するので、

$$Q_i = m_i - 1$$

となる。

すべてのノードがこの転送を行うため、1 回のフラッディングで行きかうクエリーの総数 Q は、全ピアの送信クエリー数の総和なので、

$$Q = \sum_{i=1}^n (m_i - 1) + 1 = 2m - n + 1$$

となる。ここで定数項として「+ 1」が与えられるのは、発信元のピアはすべての隣接ピアに対してクエリーを送信するためである。

以上より、フラッディングによって発生する無駄なクエリーの数は、

$$\begin{aligned} Q - Q_{min} &= (2m - n + 1) - (n - 1) \\ &= 2(m - n + 1) \end{aligned}$$

となる。今日、インターネットポロジーにおけるリンク数はべき乗則 (Power-Law) に従うことが示されており、ピアが増える割合よりリンクの数の増える割合の方がはるかに大きいので、ネットワークが拡大するにつれて無駄なクエリーは加速度的に増加してしまい、ネットワークを圧迫する原因となる。

第4章 リンクカットの手法

本章では、本研究で検討したリンクカットの手法について説明する。

4.1. ランダムリンクカット

最も単純なリンクカットは、双方向リンク、つまり互いに逆向きの二つの片方向リンクのいずれかを、ランダムにカットするものである。本稿ではこれを「ランダムリンクカット」と呼ぶ。

以下はランダムリンクカットの具体的な手法である。(図4参照)

1. 1つのリンクに対して乱数を発生させる。
2. 乱数が奇数であれば、ピア番号の小さいピアから番号の大きいピアへのリンクをカットする。
3. すべてのリンクに対して1, 2の作業を行う。

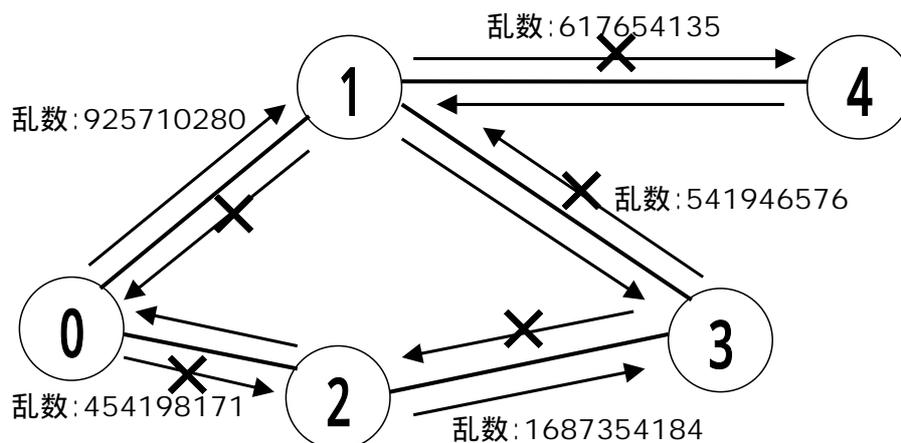


図4.ランダムリンクカットのイメージ図

4.2.出線数ベースリンクカット A

双方向リンクもしくは自分を起点とする片方向リンクで結ばれている隣接ピア数を、そのピアの「出線数」を呼ぶことにする。

一般に、出線数の大きなピアは、平均的な出線数を持つピアに比べてより早くクエリーを受信できる場合が多いと考えられる。また、出線数の大きなピアにクエリーを流しておく、より早くクエリーがネットワーク全体へ行きわたるとも考えられる。ここで本研究では、自分より出線数の少ないピアに向かうリンク以外のリンクをカットし、かつ自分の隣接ピアの中で最大の出線数を持つピアへ向かうリンクはカットしないリンクカット方式を検討した。本稿ではこれを「出線数ベースリンクカット A」と呼ぶことにする。

図5は、出線数ベースリンクカット A を用いた場合である。中央の出線数4のノードに注目すると、隣接ノードの中で出線数が2と6のノードにはクエリーを転送するが、出線数が5のノードへのリンクはカットされる。このようなリンクカットをネットワーク全体で行うことによって、クエリーの転送される量は削減される。

ここで、隣接しているリンク同士が同時刻にクエリーを受け取ってしまった場合は、ランダムリンクカットと同様に乱数を発生させ、偶数ならピア番号の大きいほうからのリンクを、奇数ならピア番号の小さいほうからのリンクをカットする。

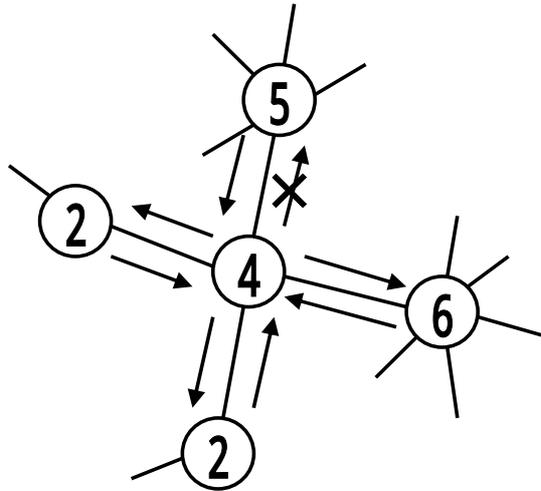


図5.出線数ベースリンクカット A でのピア0からのフラッディング

4.3.出線数ベースリンクカット B

「出線数ベースリンクカット A」とは逆に、自分より出線数の多いリンク以外のリンクをカットするリンクカット方式も検討した。本稿ではこれを「出線数ベースリンクカット B」と呼ぶことにする。

出線数ベースリンクカット Bは「出線数ベースリンクカット A」の補集合的な存在として提案しているので、隣接ピアのリンク数が自分のリンク数よりも大きいピアにのみクエリーを転送し、リンク数が「自分 = 隣接ピア」の場合は、リンクはカットされる。(図6)

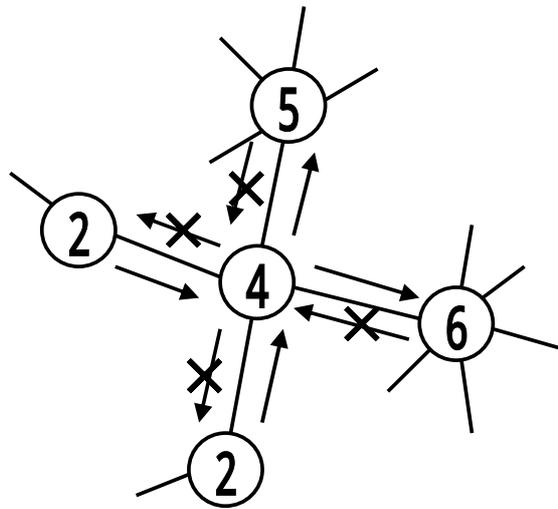


図6.出線数ベースリンクカット B のフラッディング

提案する出線数情報を用いた2つのリンクカット方式は、双方向リンクでダイレクトに連結されているピア同士が、相互に自分の出線数情報を交換し合うことで容易に実現できる。

しかし、本稿で提案した「ランダムリンクカット」「出線数ベースリンクカット A」「出線数ベースリンクカット B」の3手法すべてにおいて、その提案のみではすべてのピアにクエリーがいきわたらない場合が存在する。すべてのピアにクエリーがいきわたらなければ本研究は失敗となる。そこで、タイムアウト処理というものを用い、この問題点を処理した。タイムアウト処理については第5章で説明する。

第5章 タイムアウト処理

本研究の提案方法ではどうしてもクエリーの届かないピアが発生してしまう。それを解消するのがタイムアウト処理である。本章ではタイムアウト処理について説明する。

5.1. タイムアウト処理の原理

あるピアから送られてくるはずのクエリーが一定時間経過後転送されてこなかった場合、本来カットされている方向のピアに対してクエリーを送信するというものである。下記はタイムアウト処理が必要になる場合の例である。

ピア1からクエリーを転送する場合を考慮してみる(図7)。ピア1からクエリーを受け取ったピア2は各隣接ピアと自分の出線数の情報を比較する。ピア3は隣接ピアの中で最大の出線数を持っている。ピア4はピア2よりも出線数は大きいが出線数ではない。ピア5はピア2よりも出線数が少ない。以上のことより、ピア2はピア3とピア5にクエリーを転送することになる。しかし、ここで、ピア3とピア4、ピア4とピア5が他のピアを介してつながっていない場合、ピア4およびピア4に接続している他のピアにクエリーは届かないことになる。そこで本来はカットされているピア2からピア4へのリンクにクエリーを転送しこの問題を解消する。

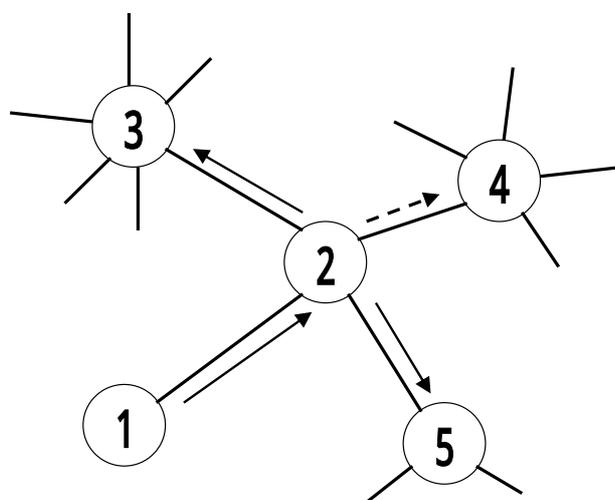


図7.タイムアウト処理の例

5.2.タイムアウト時間の設定

タイムアウト時間を適切に設定しなければ優良な実験結果は得られないことが多い。タイムアウト時間が短ければ、本来クエリーが届いてくるはずであるのに、届く前にクエリーを転送してしまう可能性があり、無駄なクエリーが増えてしまう。逆に、タイムアウト時間が長すぎると、いつまでたってもクエリーを転送せず、検索時間が大幅にかかってしまう。

そこで本研究では、以下のようなタイムアウト時間の提案をした。

- ・リンクカットしていないP2Pネットワークでのフラッディング時の各ピアから全ピアへのクエリーの最大到達時間の平均の半分とする。

最大到達時間の平均 T は、ピア数 n のネットワークで、あるピア i からクエリーを転送するときの全ピアへクエリーが到達するまでの時間を t_i としたとき、

$$T = \sum_{i=1}^n t_i$$

で表される。

本稿のシミュレーション実験での、ランダムリンクカット、出線数ベースリンクカット A 及び B では、タイムアウト時間を上式に基づいて実験を行っている。

第6章 シミュレーション実験

6.1 .BRITE

本研究を行うにあたって、仮想ネットワークの構築を行った。そのときに用いたジェネレータが BRITE[3]である。ここでは、シミュレーション実験に用いたパラメータについて説明する。

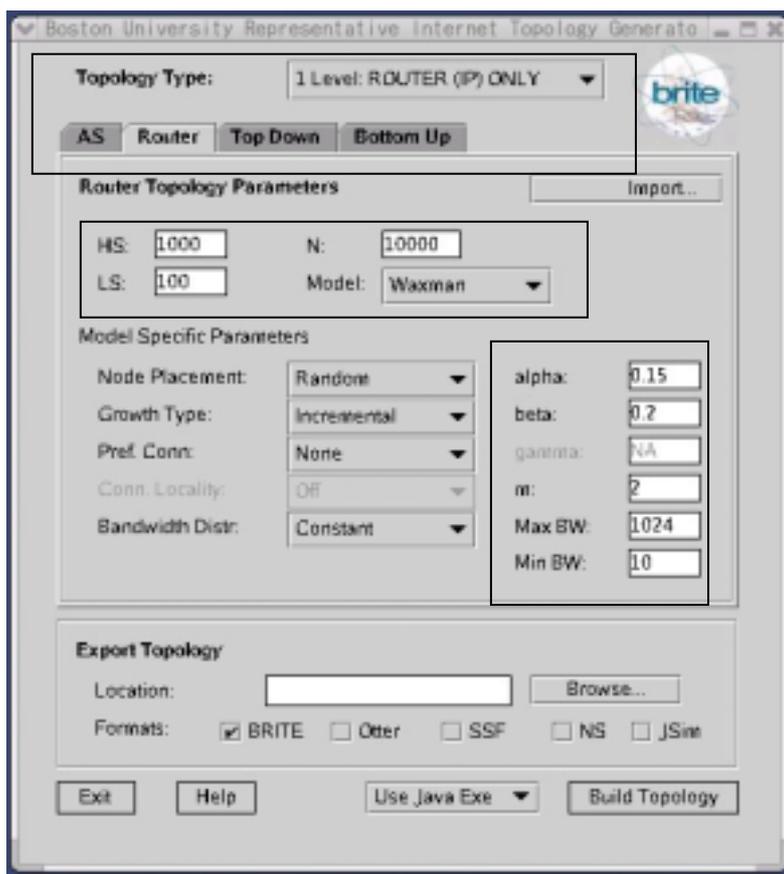


図 8.BRITE

は構築するネットワークのノードを何と置くかを定めるもので、本実験ではルータを1つのノードと見なしている。

では構築するネットワークのパラメータを設定するものである。HSは(x,y)平面座標で、HS=1000ならば、それぞれ0から1,000までの間に点がプロットされる。Nはノードの数である。Modelはネットワークのトポロジーで、本実験ではWaxman(ランダムグラフ)とBA(Power-Lawグラフ)を用いた。

はModelで選択したトポロジーのパラメータである。alpha、betaはWaxmanが提唱するWaxmanモデルで用いるパラメータで、BAのときは使用しない。mはノード数に対するリンク数で、m=2の場合はリンク数がノード数の約2倍になる。

6.1.1.Waxman モデル

B.M.Waxman が提唱したネットワーク形態である。
あるノード u 、 v の間にリンクが存在する確率は以下の式で与えられる。

$$P(\{u, v\}) = \alpha \exp \frac{-d(u, v)}{L\beta}$$

、 α はパラメータで、 $0 < \alpha < 1$ の値、 L は 2 点間の最大距離である。

6.1.2.Barabasi-Albert モデル

本稿では省略して「BA」と用いている。A.L.Barabasi と R.Albert によって提唱されたネットワークである。このネットワークは Power-Law トポロジーに従うネットワークで、現在のネットワークトポロジーに近いものとされている。

あるノード i がネットワークに接続される場合、ネットワークにすでに属しているノード j に i が接続する確率は以下の式によって与えられる。

$$P(i, j) = \frac{d_j}{\sum_{k \in V} d_k}$$

d_i : 接続しようとするノードの出線数

$\sum_{k \in V} d_k$: 既存ノードの出線数の合計

6.2.ネットワークの構築

実験を行うにあたって、BRITEによりネットワークを構築する。構築するネットワークは、IPネットワークと、P2Pネットワーク（Waxmanモデルと Waxmanモデルの2種類）である。この3つのネットワークを1セットとして、ピア数やリンク数の違うネットワークのセットをいくつか作り、それぞれを比較する。以下に、実験を行った手順（1例）を説明する。

6.2.1.IPネットワークの構築

BRITEのパラメータ

- ・トポロジ：Router
- ・HS：1000
- ・N：2000
- ・m：2
- ・Max BW：1024
- ・LS：100
- ・Model：BA（BarabasiAlbert）
- ・2
- ・Min BW：10

図9はBRITEによって構築されたネットワークのデータである。左からリンク番号、from ノード番号、to ノード番号、ノード間距離、伝播遅延、帯域幅、階層的である場合の発信元ノードのAS番号、階層的である場合の受信もとのノードのAS番号、ルーティングの分類によるリンクのタイプである。

リンク番号	from ノード番号	to ノード番号	ノード間距離	伝播遅延	帯域幅	階層的である場合の発信元ノードのAS番号	階層的である場合の受信もとのノードのAS番号	リンクのタイプ
0	415	928	340.3367215048327	1.1352210918689378	10.0	-1	-1	E_AS U
1	415	1998	432.1367672452865	1.4414330838345724	10.0	-1	-1	E_AS U
2	928	1998	758.6949321038968	2.530733885586297	10.0	-1	-1	E_AS U
3	1240	928	941.8961209801482	3.141807259888214	10.0	-1	-1	E_RT U
4	1240	415	608.8825830979238	2.8310136791297255	10.0	-1	-1	E_RT U
5	1688	1998	798.1628906432571	2.6623848243815966	10.0	-1	-1	E_RT U
6	1688	415	372.88738246285567	1.2438184234252974	10.0	-1	-1	E_RT U
7	863	1688	56.859475903318	0.189662796331314	10.0	-1	-1	E_RT U
8	863	1998	843.9840045877647	2.8152276985263076	10.0	-1	-1	E_RT U
9	1175	1240	1028.622865777346	3.431116555231507	10.0	-1	-1	E_RT U
10	1175	928	208.21623375712888	0.6945245961676096	10.0	-1	-1	E_RT U
11	1419	1240	877.8228750721888	2.9261019398769447	10.0	-1	-1	E_RT U
12	1419	1998	528.146132919285	1.7616858557505304	10.0	-1	-1	E_RT U
13	285	1175	314.8143308831621	1.647439061662951	10.0	-1	-1	E_RT U
14	285	863	645.3167778427384	2.152525057327287	10.0	-1	-1	E_RT U
15	1883	928	897.471203099836	2.895267232179123	10.0	-1	-1	E_RT U
16	1883	1688	851.7806845767284	2.8409656142614344	10.0	-1	-1	E_RT U
17	1730	285	706.4828595633861	2.3563863069586656	10.0	-1	-1	E_RT U
18	1730	863	61.91122676965643	0.2065136323969449	10.0	-1	-1	E_RT U
19	798	928	708.7858267474879	2.3641883237349752	10.0	-1	-1	E_RT U
20	798	863	610.8436788573653	2.837855193455676	10.0	-1	-1	E_RT U
21	480	1419	617.599284714195	2.860069798573008	10.0	-1	-1	E_RT U
22	480	928	902.2532903791484	3.6996930244487623	10.0	-1	-1	E_RT U
23	965	285	696.5165693557585	2.28997278295683	10.0	-1	-1	E_RT U
24	965	415	541.5579378053653	1.80644281352142379	10.0	-1	-1	E_RT U
25	481	285	480.5132672469375	1.6028197321881912	10.0	-1	-1	E_RT U
26	481	1998	817.4548868945315	2.7267349938948948	10.0	-1	-1	E_RT U
27	603	1175	528.1865200854713	1.7618405866817082	10.0	-1	-1	E_RT U
28	603	928	601.8180788244899	2.8074490293688113	10.0	-1	-1	E_RT U
29	82	480	511.5173560343651	1.7062382224250787	10.0	-1	-1	E_RT U
30	82	285	735.1720146971232	2.4529370071855476	10.0	-1	-1	E_RT U
31	1845	1883	405.1777387764535	1.351527458294009	10.0	-1	-1	E_RT U
32	1845	1998	351.6503945682416	1.1729794569822867	10.0	-1	-1	E_RT U

図9.BRITEによって構築されたネットワークの元データ

6.2.2.距離の書き換え

本研究に必要なデータは、from データ、to データ、ノード間距離のみであり、しかも本研究では物理網のノード間1ホップにつき、時間1経過することになっているため、ノード間距離を1（距離1 = 1ホップ）にする必要がある。そこで図1のデータをC言語により作成したプログラム1を用いて、from ノード、to ノード、ノード間距離1に並び替える。（図10）



図 10.ネットワークデータ（IP ネットワーク）

6.2.3.P2P ネットワークの構築

次に、この並び替えたデータから、自作プログラム 2 を用いて接続リンク数が 2 以下であるノードの数を求め、その数と同数のノードを持つネットワークを BRUTE で構築し必要データのみ書き換える (図 11)。本研究ではこのノードにピアをおく。

本研究では 1 つの IP ネットワークに対して Waxman モデルと Waxman モデルの 2 つのネットワークを構築する。

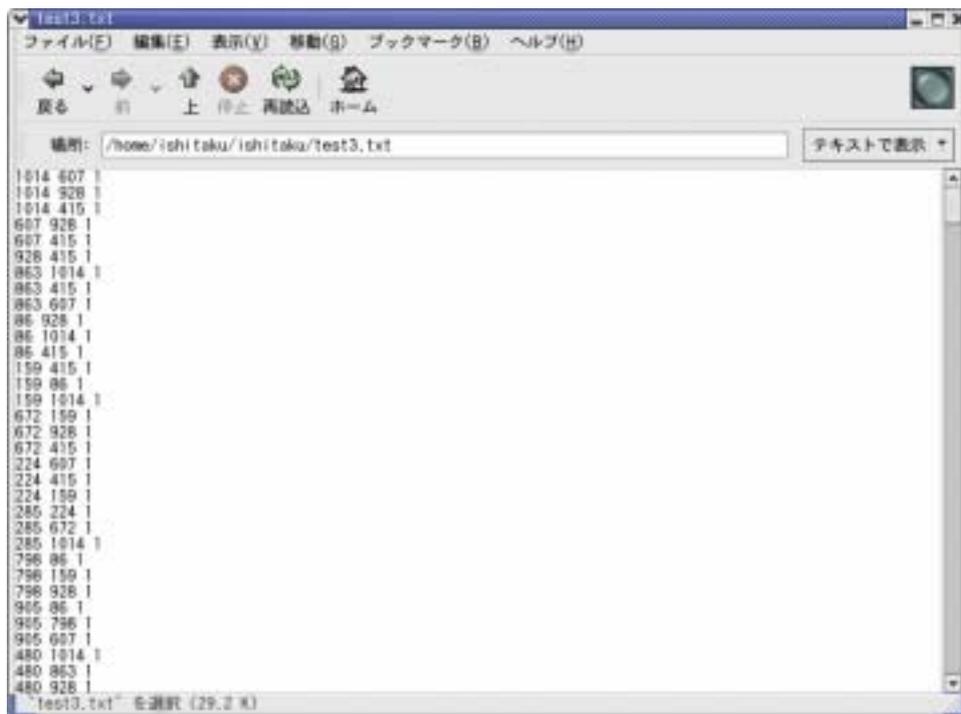


図 11.ネットワークデータ (P2P ネットワーク)

6.2.4.P2P ネットワークのピア間距離の設定

本研究で用いている IP ネットワークと P2P ネットワークは以下の図 1 2 のような関係になっている。

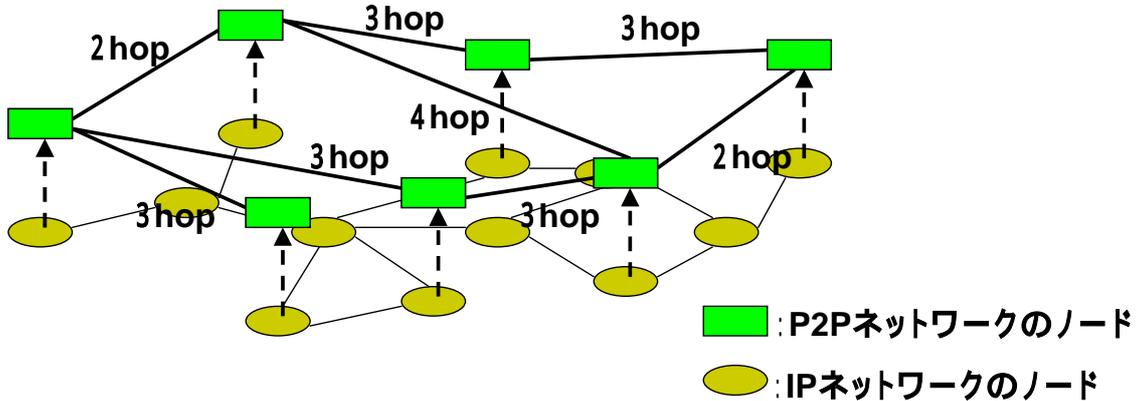


図 12.IP ネットワークと P2P ネットワークの関係図

P2P ネットワークで 1 hop のところを、IP ネットワークでは数 hop であることがわかる。このことから、図 1 1 のピア間 hop 数 1 を、IP ネットワークでの hop 数に置き換えるひつようがある。図 1 3 は P2P ネットワークでのピア間 hop 数を IP ネットワークの hop 数に置き換えたものである。

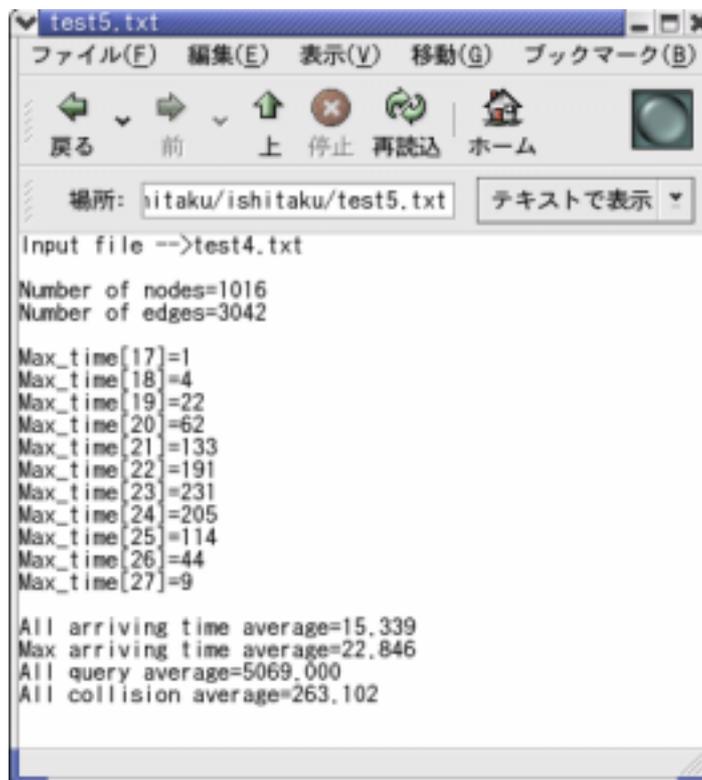


図 13.距離を置き換えた P2P ネットワークデータ

6.2.5.シミュレーション結果データ

図14は実験の結果データである。

All arriving time average : 各ピアがクエリーを受け取るまでの時間の平均
Max arriving time average : 全てのピアがクエリーを受け取るまでの時間
All query average : 総クエリー数の平均
All collision average : 衝突回数の平均



The image shows a screenshot of a text editor window titled 'test5.txt'. The window has a menu bar with 'ファイル(F)', '編集(E)', '表示(V)', '移動(G)', and 'ブックマーク(B)'. Below the menu bar is a toolbar with icons for '戻る', '前', '上', '停止', '再読込', and 'ホーム'. The status bar shows the file path '場所: hitaku/ishitaku/test5.txt' and a dropdown menu set to 'テキストで表示'. The main text area contains the following simulation results:

```
Input file -->test4.txt
Number of nodes=1016
Number of edges=3042
Max_time[17]=1
Max_time[18]=4
Max_time[19]=22
Max_time[20]=62
Max_time[21]=133
Max_time[22]=191
Max_time[23]=231
Max_time[24]=205
Max_time[25]=114
Max_time[26]=44
Max_time[27]=9
All arriving time average=15.339
Max arriving time average=22.846
All query average=5069.000
All collision average=263.102
```

図14.シミュレーション結果データ

6.3.シミュレーション結果

本シミュレーションは2つのトポロジーを持つネットワークを用い、IPネットワークのピア数・リンク数、P2Pネットワークのピア数・リンク数をそれぞれ変え、ネットワークの規模、リンク数、トポロジーによって、本提案がどのような結果をもたらすかを検証した。

6.3.1.シミュレーションに用いたデータ

Case 1、2、3は、ピア数とリンク数の割合は同じで、ネットワークの規模の大きさによる比較をする。Case 3、4、5は、IPネットワークは同じだがP2Pネットワークのピア数とリンク数の割合を変え、リンク数による比較をする。

また、それぞれについて Waxman モデル・BA モデルの2つを作り、トポロジーによる比較も行う。

表 1.Case 1

階層	IP ネットワーク	P2P ネットワーク	
トポロジー	BA モデル	Waxman モデル	BA モデル
ピア数	200	95	95
リンク数	400	285	279

表 2.Case 2

階層	IP ネットワーク	P2P ネットワーク	
トポロジー	BA モデル	Waxman モデル	BA モデル
ピア数	1000	493	493
リンク数	2000	1479	1473

表 3. Case 3

階層	IP ネットワーク	P2P ネットワーク	
トポロジー	BA モデル	Waxman モデル	BA モデル
ピア数	2000	998	998
リンク数	4000	2994	2988

表 4. Case 4

階層	IP ネットワーク	P2P ネットワーク	
トポロジー	BA モデル	Waxman モデル	BA モデル
ピア数	2000	998	998
リンク数	4000	1996	1933

表 5. Case 5

階層	IP ネットワーク	P2P ネットワーク	
トポロジー	BA モデル	Waxman モデル	BA モデル
ピア数	2000	998	998
リンク数	4000	3992	3982

6.3.2.各 Case の結果

Case1 : ピア数 95

表 6.Waxman モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	8.962	11.41	10.84	17.53
平均最大到達時間	14.03	20.62	17.34	31.33
リンクカット無しのクエリー数	476	476	476	476
最低クエリー数	94	94	94	94
平均クエリー数	476	293.5	334.5	281.8
リンクカット無しとの比	1	0.6167	0.7027	0.592
最低クエリー数との比	5.064	3.123	3.558	2.998

表 7.BA モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	8.969	11.39	10.83	17.13
平均最大到達時間	13.58	21.74	16.15	28.54
リンクカット無しのクエリー数	464	464	464	464
最低クエリー数	94	94	94	94
平均クエリー数	464	288.3	329.2	286.32
リンクカット無しとの比	1	0.6213	0.7094	0.6171
最低クエリー数との比	4.936	3.067	3.502	3.046

Case2 : ピア数 493

表 8.Waxman モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	14.94	20.49	18.03	32.63
平均最大到達時間	22.48	37.47	30.99	56.21
リンクカット無しのクエリー数	2466	2466	2466	2466
最低クエリー数	492	492	492	492
平均クエリー数	2466	1505.1	1728.1	1429.6
リンクカット無しとの比	1	0.6104	0.7008	0.5797
最低クエリー数との比	5.012	3.059	3.512	2.906

表 9.BA モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	13.28	17.81	15.24	27.38
平均最大到達時間	19.85	33.35	27.08	44.73
リンクカット無しのクエリー数	2454	2454	2454	2454
最低クエリー数	492	492	492	492
平均クエリー数	2454	1506.9	1697.7	1472.6
リンクカット無しとの比	1	0.6141	0.6918	0.6001
最低クエリー数との比	4.988	3.063	3.451	2.993

Case3 : ピア数 998、リンク数 約 3000

表 10.Waxman モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	17.65	24.51	21.13	39.73
平均最大到達時間	26.11	45.78	32.76	66.01
リンクカット無しのクエリー数	4991	4991	4991	4991
最低クエリー数	997	997	997	997
平均クエリー数	4991	3037.6	3526.4	2913.1
リンクカット無しとの比	1	0.6086	0.7066	0.5837
最低クエリー数との比	5.006	3.047	3.537	2.922

表 11.BA モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	15.22	20.28	17.3	31.66
平均最大到達時間	22.54	37.25	26.28	54.39
リンクカット無しのクエリー数	4979	4979	4979	4979
最低クエリー数	997	997	997	997
平均クエリー数	4979	3052.5	3423.9	2983
リンクカット無しとの比	1	0.6131	0.6877	0.5991
最低クエリー数との比	4.994	3.062	3.434	2.992

Case4 : ピア数 998、リンク数 約 2000

表 12.Waxman モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	22.32	37.28	26.6	62.13
平均最大到達時間	33.49	70.31	47.39	113.8
リンクカット無しのクエリー数	2995	2995	2995	2995
最低クエリー数	997	997	997	997
平均クエリー数	2995	2023.2	2330.8	1846.4
リンクカット無しとの比	1	0.6755	0.7782	0.6165
最低クエリー数との比	3.004	2.029	2.338	1.852

表 13.BA モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	18.44	28.35	20.48	43.89
平均最大到達時間	27.97	57.82	37.77	77.14
リンクカット無しのクエリー数	2989	2989	2989	2989
最低クエリー数	997	997	997	997
平均クエリー数	2989	2031.2	2304.3	1960.9
リンクカット無しとの比	1	0.6796	0.7709	0.656
最低クエリー数との比	2.998	2.037	2.311	1.967

Case5：ピア数 998、リンク数 約 4000

表 14.Waxman モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	15.52	20.1	19	31.73
平均最大到達時間	22.59	36.65	28.82	53.53
リンクカット無しのクエリー数	6987	6987	6987	6987
最低クエリー数	997	997	997	997
平均クエリー数	6987	4049.4	4597.9	3965.7
リンクカット無しとの比	1	0.5796	0.6581	0.5676
最低クエリー数との比	7.008	4.062	4.612	3.978

表 15.BA モデル

	リンクカット 無し	ランダムリンク カット	出線数ベース リンクカット A	出線数ベース リンクカット B
平均到達時間	13.91	17.64	16.04	27.21
平均最大到達時間	20.18	32.8	23.79	42.41
リンクカット無しのクエリー数	6967	6967	6967	6967
最低クエリー数	997	997	997	997
平均クエリー数	6967	4061.2	4473.2	4025.8
リンクカット無しとの比	1	0.5929	0.6421	0.5778
最低クエリー数との比	6.988	4.073	4.487	4.038

第7章 シミュレーション結果による考察

各ケースにおける比較

Case1,2,3 の比較

この3つのデータの比較は、ピア数とリンク数の割合は同じで、単にネットワークの規模によって、本提案にどのような結果が出るのかを検証しようとしたものである。

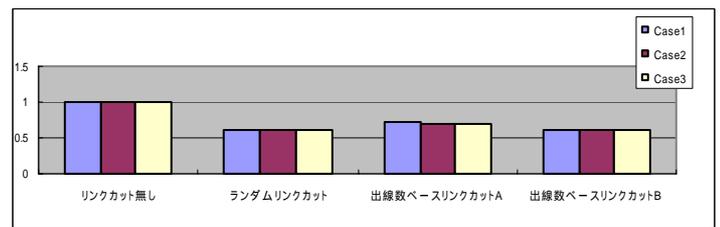
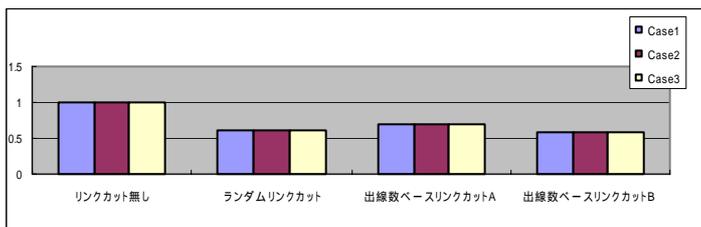


図 15.クエリー数のカット無しとの比 (左 : Waxman モデル、右 : BA モデル)

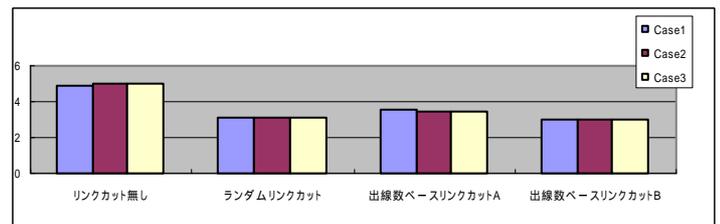
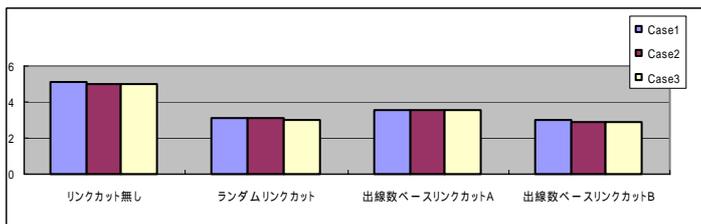


図 16.クエリー数の最低クエリー数との比 (左 : Waxman モデル、右 : BA モデル)

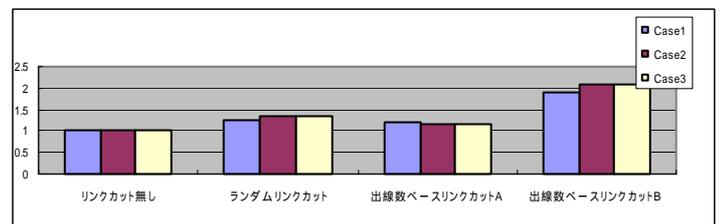
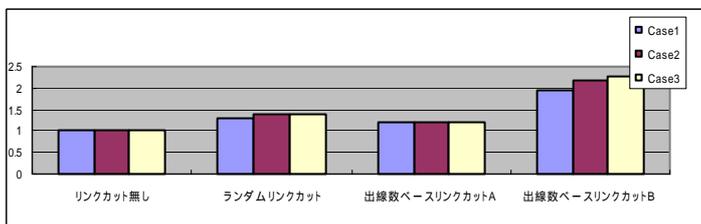


図 17.平均到達時間のカット無しとの比 (左 : Waxman モデル、右 : BA モデル)

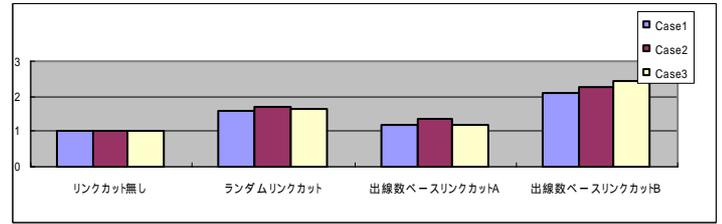
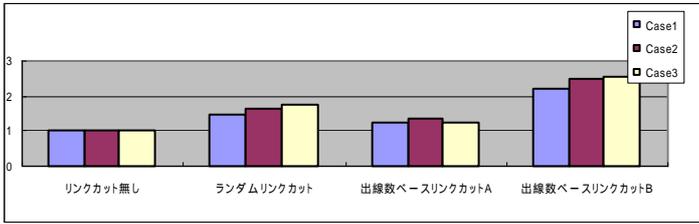


図 18.平均最大到達時間のカット無しとの比（左：Waxman モデル、右：BA モデル）

Case 1 ~ 3 の比較では、ネットワークの規模に関して大きな差が出ることはなかった。クエリーの削減率はネットワークの大きいほうがやや高く、到達時間はネットワークの小さいほうがやや速くなった。

このことから、ネットワークの規模に関しては、本提案では差が出ず、どのような規模のネットワークに対しても汎用的であることが推察される。

Case3,4,5 の比較

この3つの比較は、ネットワークのピア数とリンク数の割合（リンクの密度）によってどのように結果が変わるのかを検証したものである。

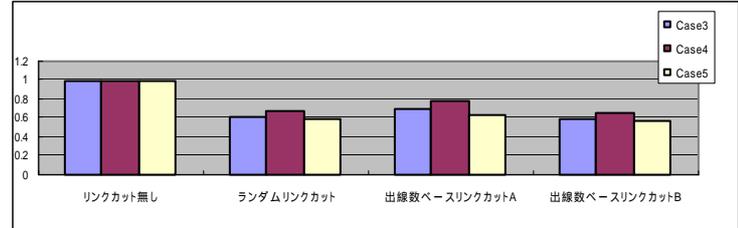
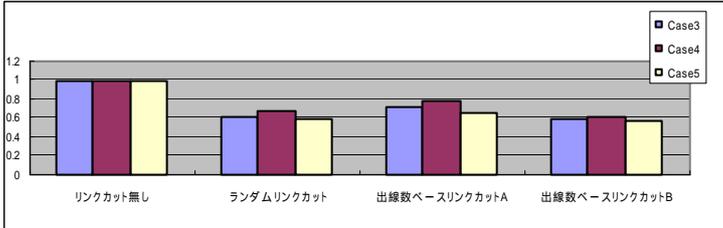


図 19.クエリー数のカット無しとの比（左：Waxman モデル、右：BA モデル）

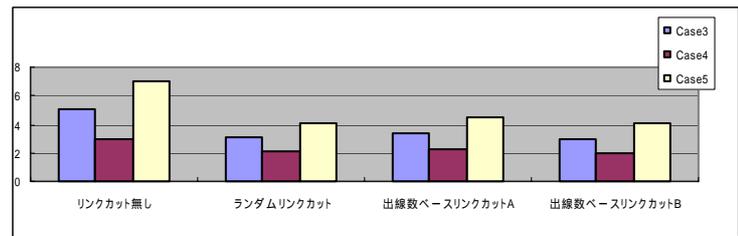
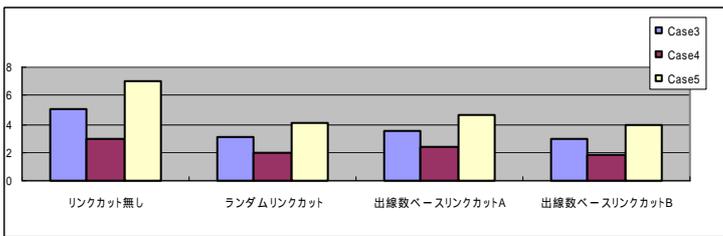


図 20.クエリー数の最低クエリー数との比（左：Waxman モデル、右：BA モデル）

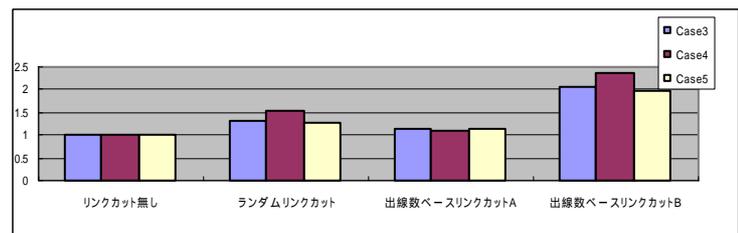
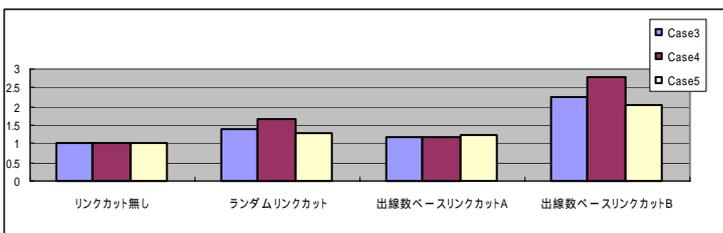


図 21.平均到達時間のカット無しとの比（左：Waxman モデル、右：BA モデル）

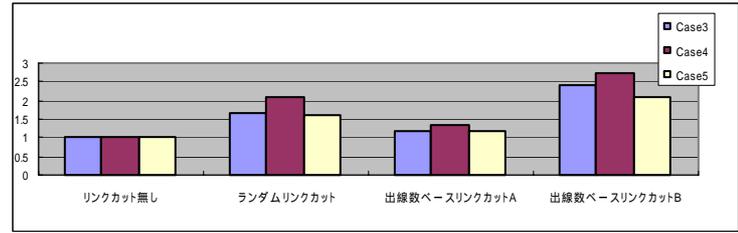
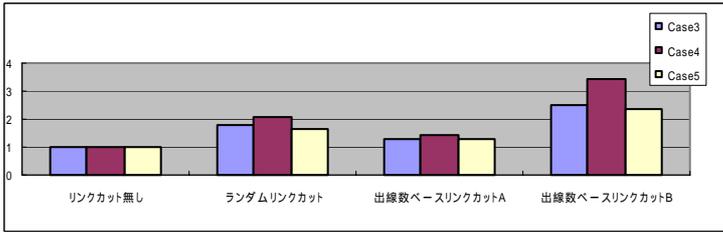


図 22.平均最大到達時間のカット無しとの比 (左：Waxman モデル、右：BA モデル)

Case 3 ~ 5 の比較では、より顕著な結果が出た。ネットワークのリンク密度が大きければ大きいほどクエリー数の削減率は大きくなり、また、到達時間も短くなる。リンク密度が大きいほど到達時間が短くなるのは、より多くの経路が存在するため、より最短な経路でクエリーが転送されるためである。

この比較により、リンク密度が大きいほど本提案は有効であることがわかった。

第 8 章 まとめ・結論

本提案に対しさまざまなネットワークでのシミュレーションを行った結果、

- ネットワークの規模に関してはあまり影響を受けない。
- ネットワークのトポロジー（Waxman モデルと BA モデル）に関してはあまり影響を受けない。
- ネットワークのリンク密度に関しては、密度の大きいネットワークほど本提案が効果的である。
- 本提案（ランダムリンクカット、出線数ベースリンクカット A、B）の比較では、クエリー数の削減率のみを取ればランダムグラフや出線数ベースリンクカット B の結果のほうが良いものであるが、クエリーが到達するまでの時間が長く、検索に時間がかかってしまう。このことより、総合的に見れば、出線数ベースリンクカット A の提案がより効率的である。

以上のことがわかった。

本提案は、どのような種類のトポロジー・規模のネットワークに対しても汎用的であり、効率的なものであると推察する。

謝辞

本研究を行うにあたり、ご指導いただきました塩田茂雄助教授に深く感謝いたします。

また、本研究室の先輩、同学年の皆様、池田研究室の方々にも様々な助言、励ましの言葉などいただき、大変感謝しております。

1年間、ありがとうございました。

参考文献

- [1] H.Tangmunarunkit, R.Govindan, S.Jamin, S.Shenker, and W.Willinger, “Network topology generators: degree-based vs. structural”, ACM SIGCOMM 2002, pp.147-159.
- [2] B.M.Waxman, “Routing of multipoint connections”, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL.6, NO.9, DECEMBER 1988, pp.1617-1622.
- [3] A.Medina, A.Lakhine, I.Matta, J.Byers, “BRITE: universal topology generation from a user’s perspective”, Computer science department, Boston university, BUCS-TR-2001-002, April 12, 2001.
- [4] Paul R. Guimaraes Jr, Marcus A. M . de Aguiar, Jordi Bascompte, Pedro Jordano and Sergio Furtado dos Reis, “Random initial condition in small Barabasi-Albert networks and deviations from the scale-free behavior”, Instituto de Biologia, Universidade Estadual de Campinas (UNICAMP), Caixa Postal 6109, 13083-970 Campinas, Brazil, Instituto de Fisica ‘Gleb Wataghin’, Unibersidade Estadual de Campinas (UNICAMP), Caixa Postal 6165, 13083-970 Campinas, Brazil, Intaegrative Ecology Group, Estacion Biologica de Donana, CSIC, Apdo. 1056, E^41080 Sevilla, Spain, Departamento de Parasitologia, Instituto de Biologia, Universidade Estadual de Campinas (UNICAMP), Caixa Postal 6109, 13083-970, Cmpinas, SP, Brazil, (Dated: January 11, 2005)
- [5] ラディア・パールマン著、加藤 朗 監訳「インターコネクションズ」翔泳社
- [6] A.L.Barabasi and R.Albert, “Emergence of Scaling in Random Networks”, Science, pp.509-512, October 1999.